Self-adaptation supporting multimedia interaction in large-scale real-time systems

L. Provensi[1], F. Eliassen[1], and R. Vitenberg[1]

[1]Department of Informatics, University of Oslo, PO Box 1080 Blindern, N-0316 Oslo, Norway
Contact Email: {provensi,frank,romanvi}@ifi.uio.no

The Verdione project aims to provide a framework for development of distributed interactive multimedia systems. The objective is to allow people who are distributed around the globe to interact with each other as though they were co-located. Our research is specifically concerned about *self-adaptation* aspects of the project. Self-adaptation is the ability of a software system to automatically evaluate its internal state and external operation environment and modify its own behavior accordingly [1]. The adaptation behavior is specified as high-level objectives, stating what actions should be taken at run-time to automatically optimize the system, recover from failures, protect from threats, etc.

Although self-adaptation is a well-known idea, there is no known mechanisms that can perform adaptations efficiently considering the characteristics of the applications targeted by Verdione. One example is the *World Opera* application, which will enable the realization of distributed operas in which singers and musicians from opera houses world-wide participate. In this application, we have to face the challenge of many-to-many dissemination of multimedia data to a highly heterogeneous and dynamic large-scale set of content consumers. Considering this challenge, the objective of our research is to develop novel mechanisms for self-adaptation, that are flexible, dependable and scalable. Our solution is based on two new features that can be integrated to the self-adaptation mechanism:

1. **Feedback Control Loop (FCL) design model** – A self-adaptive software system can be described as a closed-loop mechanism, where an adaptation engine receives feedback from the software system and from its environment, and adjusts the system accordingly. The control aspects of self-adaptive systems are of key importance during the design phase, and thus FCL governing the adaptation should be treated as a first class entity [2]. This design can improve the flexibility, because the implementation of each control function is not fixed and can be selected at design or run-time. The scalability is also improved, because each function can be independently deployed in different locations in a large-scale systems.
2. **Quality of the adaptation** – The self-adaptation mechanism is also a software system that can have a set of qualitative requirements, such as constraints for resource consumption and execution time. The self-adaptation mechanism should, itself, be adapted according to variations in its quality levels. This feature can improve the dependability of the self-adaptation solution.

The first step of our research is focused on the first feature. We are investigating the means of designing the self-adaptive mechanism as a FCL entity composed of control functions that can be independently implemented and deployed, and that can also expose qualitative information about its internal operation.

References:

[1] P. Oreizy et al. An architecture-based approach to self-adaptive software. IEEE Intelligent Systems, **14(3)**:54–62, 1999.
[2] H. Müller et al. Visibility of control in adaptive systems. In Proceedings of the 2nd international workshop on Ultra-large-scale software-intensive systems, 2008.